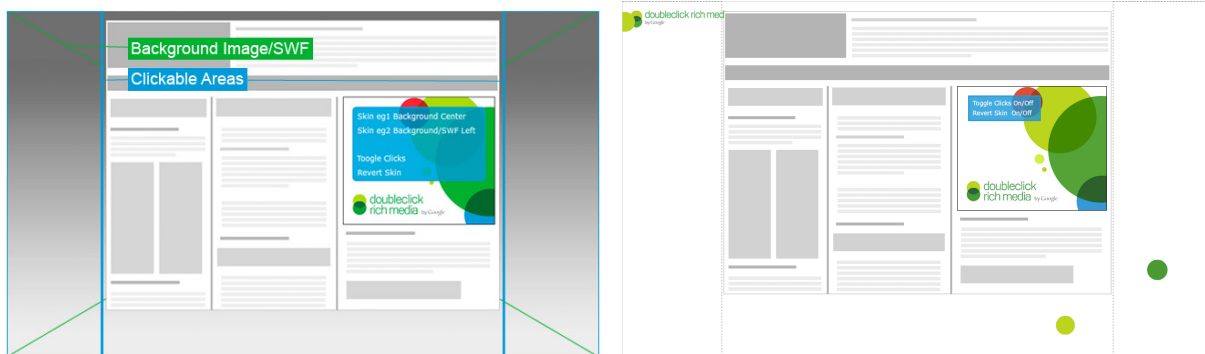


# Clickable Skin

## Introduction

Use the Clickable Skin to grab a users attention by taking your creative content and replace the entire background of the publisher's page. This easy to use component introduces a range of possibilities for a campaigns display, and unlike backgrounds and skins of the old days this one delivers the complete Rich Media experience with animation and interactivity, bringing a whole host of new possibilities to the format.



Demos of the Clickable Skin and components can be found under [Rich Media Gallery - Background Skin](#)

## Build & Delivery

The Clickable Skin takes creative content specified by you and uses it to replace the entire background of the publisher's page.

**NOTE:** If you want additional advice around a concept you may have or extra help using this component at any stage please contact our Rich Media Technical Services support team at [dclk-drmtechnical@google.com](mailto:dclk-drmtechnical@google.com)

The Clickable Skin component available for both ActionScript 3 and ActionScript 2 and is supported within DoubleClick Studio for **ALL** Rich Media formats:

> In-Page

> Floating

> Multi-floating

> In-Page with Floating

> Expanding

> Peel-Down

For more specific documentation relating to these formats please visit our [help center](#).

**NOTE:** The In-Page setup requires JavaScript tags, if iFrame tags are used please setup as an Expanding.

**IMPORTANT:** Animated Skins are only supported with the 'In-Page with Floating' option, where the animated SWF Skin is setup as the Floating.

Template files are available to assist in the creative build and can be downloaded below:

> AS3 Clickable Skin Component Version 2 - <http://goo.gl/ts0K2>

This contains an .mxp package file that will install the component into Flash CS4 or higher.

> AS2 Clickable Skin Component Version 2 - <http://goo.gl/nyse7>

This contains an .mxp package file that will install the component into Flash CS4 or higher.

> AS3 Clickable Skin Component Templates - <http://goo.gl/AxYZU>

This contains example files for a 1x1 Floating creative and Skin, a 300x250 In-Page creative with toggle button and Skin, and a 300x250 In-Page with Float creative and Flash Skin.

> AS2 Clickable Skin Component Templates - <http://goo.gl/aUYae>

This contains example files for a 1x1 Floating creative and Skin, a 300x250 In-Page creative with toggle button and Skin, and a 300x250 In-Page with Float creative and Flash Skin.

# Using The Component

1. Download the [AS3 Component](#) and [AS2 Component](#).

2. Once downloaded, install the component by double-clicking the downloaded .mxp file and accepting the extension disclaimer when the Adobe Extension Manager opens.

**NOTE:** Alternatively if you open Flash and go to the Help Menu, you can select Manage Extensions from the dropdown list, then navigate to the mxp you downloaded and select it to install it.

3. Open Flash & start a new AS3 or AS2 project.

4. Building the Clickable Skin in Flash.

> Place and initialise an instance of the [DoubleClick Studio Library Enabler](#) component.

> Place an instance of the 'Clickable Skin' component (found within 'DoubleClick Custom Components' folder) inside the main creative.

> If using a Flash Skin, place and initialise a second instance of the [DoubleClick Studio Library Enabler](#) component inside the floating creative.

**NOTE:** This will allow you to use DoubleClick Enabler features, such as Exit Events, within the Flash Skin.

5. Click on the 'Clickable Skin' and navigate to the Flash Properties window (CTRL+F3 or ⌘+F3) then go to the Component Parameters section. This is where you can make changes to how the component will behave and react within your creative. Detailed information on each of the settings are below.

COMPONENT PARAMETERS	
Property	Value
Clickable Left Active	<input checked="" type="checkbox"/>
Clickable Right Acti...	<input checked="" type="checkbox"/>
Clickable Top Active	<input checked="" type="checkbox"/>
Image	<input type="text"/>
Image Attachment	fixed   ▾
Image Background ...	■
Image Position	center-top   ▾
Image Repeat	no-repeat   ▾
Start Active	<input checked="" type="checkbox"/>
SWF	<input type="text"/>

> **Clickable Left Active:** Boolean - If checked this specifies that the left area of the publishers site will be clickable

**NOTE:** To specify a clickable width please refer to the API settings in this document

> **Clickable Right Active:** Boolean - If checked this specifies that the right area of the publishers site will be clickable

**NOTE:** To specify a clickable width please refer to the API settings in this document

> **Clickable Top Active:** Boolean - If checked this specifies that the top area of the publishers site will be clickable

**NOTE:** To specify a clickable height please refer to the API settings in this document

> **Image:** String - This is where you specify the background skin filename. This file should be uploaded to studio along with the rest of your creatives files.

**NOTE:** This should be the full JPG filename E.g. background.jpg

> **Image Attachment:** Dropdown selection - Select if your Skin will be fixed or scroll in the browser window.

> **Image Background:** Selection - Choose the color of the site background which will display behind your Skin.

> **Image Position:** Dropdown selection - Choose the position of the Skin.

> **Image Repeat:** Dropdown selection - Select a repeat option.

**NOTE:** We recommend using 'repeat' for top-left positions, and 'no-repeat' for top-center positions.

> **Start Active:** Boolean - Specify if the Skin will start on publisher site load.

**NOTE:** If not active use the 'Clickable Skin Toggle Button' to toggle Skin

> **SWF:** String - Specify your Flash SWF Skin filename (leave blank if no 'Flash Skin' ), this file will be uploaded to Studio as a '**Floating**' creative along with the '**In-Page**' creative containing this component.

## API Usage

Dynamically create a Skin to assign more advanced features not available in the Component. Each time an instance is added it'll replace any current active instance on the publishers site.

### > ActionScript 3

```
import com.google.ads.studio.strat.clickableskin.proxy.WallPaper;
var skincomponent:WallPaper = new WallPaper();
skincomponent.proxy.backgroundImage = '';
skincomponent.proxy.addEventListener("ready", readyHandler);
addChild(skincomponent);
```

To Toggle the current active skin on/off and revert to default site skin use `revertSkinComponent()`.

```
import com.google.ads.studio.strat.clickableskin.proxy.WallPaperRevert;
var skincomponentRevert:WallPaperRevert = new WallPaperRevert();
skincomponentRevert.proxy.revertSkinComponent();
```

To Toggle clicks on/off use `toggleSkinComponent()`.

```
import com.google.ads.studio.strat.clickableskin.proxy.WallPaperToggle;
var skincomponentToggle:WallPaperToggle = new WallPaperToggle();
skincomponentToggle.proxy.toggleSkinComponent();
```

### > ActionScript 2

```
import com.google.ads.studio.strat.clickableskin.WallPaper;
var skincomponent:WallPaper = new WallPaper();
skincomponent.backgroundImage = '';
skincomponent.generateSkinComponentReady();
addChild(skincomponent);
```

To Toggle the current active skin on/off and revert to default site skin use `revertSkinComponent()`.

```
skincomponent.revertSkinComponent();
```

To Toggle clicks on/off use `toggleSkinComponent()`.

```
skincomponent.toggleSkinComponent();
```

# Advanced API Usage

Other properties you can assign to 'skincomponent.proxy' (AS3) or 'skincomponent' (AS2) are...

> **siteWidth**:Number - Specify width of publisher's main content area. If '0' the skin will attempt to automatically detect width.

**IMPORTANT:** Set this if the skin is not correctly clicking when viewed on the publisher's site.

```
//valid values: Number (Default Value: 0)
siteWidth = 0;
```

> **appendTo**:String - Specify the ID of a <div> container to load the skin into (by default this will be set on the publisher's <body>).

```
//valid values: String (Default Value: '')
appendTo = '';
```

> **clickableLeftOn**:Boolean - Whether the left section of the publishers site is clickable or not.

Example API Usage:

```
//valid values: true | false (Default Value: true)
clickableLeftOn = true;
```

> **clickableLeftTitle**:String - Set alt text title of left clickable area.

Example API Usage:

```
//valid values: String (Default Value: 'Advert - Click here')
clickableLeftTitle = 'Advert - Click here';
```

> **clickableLeftExitLink**:String - Set specific 'Exit Events' for left clickable area.

**NOTE:** This Exit Event should be named the same in DoubleClick Studio.

Example API Usage:

```
//valid values: String (Default Value: 'Skin Exit')
clickableLeftExitLink = 'Skin Exit';
```

> **clickableLeftWidth**:Number - Set left clickable area, '0' will set the entire area as clickable.

Example API Usage:

**NOTE:** The siteWidth value needs to be set for this to work correctly.

```
//valid values: Number (Default Value: 0)
clickableLeftWidth = 0;
```

> **clickableRightOn**:Boolean - Whether the right section of the publishers site is clickable or not.

Example API Usage:

```
//valid values: true | false (Default Value: true)
clickableRightOn = true;
```

> **clickableRightTitle**:String - Set alt text title of right clickable area.

Example API Usage:

```
//valid values: String (Default Value: 'Advert - Click here')
clickableRightTitle = 'Advert - Click here';
```

> **clickableRightExitLink**:String - Set specific 'Exit Events' for right clickable area.

**NOTE:** This Exit Event should be named the same in DoubleClick Studio.

Example API Usage:

```
//valid values: String (Default Value: 'Skin Exit')
clickableRightExitLink = 'Skin Exit';
```

> **clickableRightWidth**:Number - Set right clickable area, '0' will set the entire area as clickable.

**NOTE:** The siteWidth value needs to be set for this to work correctly.

Example API Usage:

```
//valid values: Number (Default Value: 0)
clickableRightWidth = 0;
```

> **clickableTopOn**:Boolean - Whether the top section of the publishers site is clickable or not.

Example API Usage:

```
//valid values: true | false (Default Value: true)
clickableTopOn = true;
```

> **clickableTopTitle**:String - Set alt text title of top clickable area.

Example API Usage:

```
//valid values: String (Default Value: 'Advert - Click here')
clickableTopTitle = 'Advert - Click here';
```

> **clickableTopExitLink**:String - Set specific 'Exit Events' for top clickable area.

**NOTE:** This Exit Event should be named the same in DoubleClick Studio.

Example API Usage:

```
//valid values: String (Default Value: 'Skin Exit')
clickableTopExitLink = 'Skin Exit';
```

> **clickableTopHeight**:Number - Set top clickable area, '0' will set the entire area as clickable.

Example API Usage:

```
//valid values: Number (Default Value: 0)
clickableTopHeight = 0;
```

Specify background properties.

> **backgroundImage**:String - Specify the background skin filename.

Example API Usage:

```
//valid values: String (Default Value: '')
backgroundImage = '';
```

> **backgroundSWF**:String - Specify your Flash SWF Skin filename (leave blank if no 'Flash Skin').

Example API Usage:

```
//valid values: String (Default Value: '')
backgroundSWF = '';
```

> **backgroundAttachment**:String - Specify if your Skin will be fixed or scroll in the browser window.

Example API Usage:

```
//valid values: String (Default Value: 'fixed'. Other Values: 'scroll')
backgroundAttachment = 'fixed';
```

> **backgroundColour**:String - Set color to display behind your Skin image.

Example API Usage:

```
//valid values: String (Default Value: '#000000')
backgroundColour = '#000000';
```

> **backgroundPositionX**:String - Set X position of the Skin image.

Example API Usage:

```
//valid values: String (Default Value: 'center'. Other Values: 'left', 'right')
backgroundPositionX = 'center';
```

> **backgroundPositionY**:String - Set Y position of the Skin image.

Example API Usage:

```
//valid values: String (Default Value: 'top'. Other Values: 'middle', 'bottom')
backgroundPositionY = 'top';
```

> **backgroundRepeat**:String - Set repeat option of Skin image.

Example API Usage:

```
//valid values: String (Default Value: 'no-repeat'. Other Values: 'repeat', 'repeat-
x', 'repeat-y')
backgroundRepeat = 'no-repeat';
```

> **backgroundStart**:Boolean - Specify if the Skin will display on publisher site load.

Example API Usage:

```
//valid values: Boolean (Default Value: true, Other Values: false)
backgroundStart = true;
```

# Setting Up Your Creative in Studio

**Option 1** - upload your image Skin creative in DoubleClick Studio.

> Setup your creative (depending on the **Regular Process** for that format).

> Navigate to the Upload tab, and upload the Skin image(s) you would like to use along with the creative containing this component (SWF).

**NOTE:** Image formats Supported: jpeg, gif, png.

> To create skin for specific resolutions, upload the resolution specific images along with the main image, naming the resolution specific images with an `_size` e.g `'..._1024.jpg', ..._1920.jpg'`.etc.

**Option 2** - upload your Flash Skin creative in DoubleClick Studio.

> Setup your creative as an **'In-Page with Floating'**.

> Navigate to the Upload tab, and upload both the Flash Skin (SWF) you would like to use, your background image Skin and creative containing this component (SWF).

**NOTE:** Remember to include an instance of the Enabler in both SWF creatives.

**1.** Delivering & testing in DoubleClick Studio.

> Navigate to the Events tab and add an 'Exit Event' called **'Skin Exit'**, with accompanying URL.

> Navigate to the Preview tab and check your Skin displays as expected.

> **If using a Flash Skin, turn off the initial load of the floating creative** by clicking 'Display Options -> Floating' and selecting start time 'None'.

**2.** When ready navigate to the Publish tab and submit to QA.

## Best Practises

- > A Skin image **must** still be used as backup even if you are using a Flash Skin. If no background image is set the Flash Skin may not display correctly. Your Flash Skin will set itself to the size of the background image dimensions.
- > If you would like to load an image Skin on its own, follow the Build Guide and setup a 'Floating' creative of size 1px x 1px.
- > The Component instance **must** always be included within the main creative SWF file **not** the background Skin SWF file. Only have one setup of the Skin build.
- > If you're using a Flash Skin set up as a floating creative **it is recommended you turn off the initial load of the floating creative** by clicking the 'Preview Tab' in Studio, navigating to 'Display Options -> Floating' and selecting start time 'None'.
- > We recommend working with a background Skin image of 1920px x 1200px.

**IMPORTANT:** Check Studio has not assigned your skin as the 'backup image', this will prevent the skin showing.

## Additional Information

- > Use the API to specify an append to <div>, if you do not want to fill the entire background on the publisher's site an append to <div> id can be specified instead.
- > Use the API to specify a site width, this will resolve any issues with the Skin not clicking on a complex publisher's site.
- > Component parameters can also be set in DFA, useful for quickly fixing issues on the publisher's site by overriding any parameters set in the component. Set these in DFAs 'Update Creative' screen within the 'Key=Value Parameters' field (just under Survey URL). Refer to the 'Advanced API' section above for properties to set.

## Information & Dependencies

Minimum flash version 9, component versions available for ActionScript 2 and 3. The component is setup to work with the [DoubleClick Studio Library Enabler](#).

## Tracking

These events will display in the 'Output Console' within the Studio Preview screen.

- > Skin Image Loaded - To record add a Studio counter for "Skin Image Loaded".
- > Skin SWF Loaded - To record add a Studio counter for "Skin SWF Loaded".
- > Toggle Clicks On/Off - To record add a Studio counters for "Toggle Skin On" and "Toggle Skin Off".
- > Toggle Skin On/Off - To record add a Studio counters for "Revert Skin On" and "Revert Skin Off".
- > Skin Exit - Add this exit event to control the URL fired when the user clicks to exit.

## Resources

- > <http://richmediagallery.com/clickable-skin>
- > [AS3 FLA]
- > [AS2 FLA]
- > [AS3 Component]
- > [AS2 Component]
- > DoubleClick Rich Media Help Centre
- > DoubleClick Rich Media Studio
- > DoubleClick Rich Media Gallery
- > [@rmgallery](#) - DoubleClick Rich Media Gallery Twitter Account

## Contact Us

If you still have any questions relating to this format or build then please feel free to contact our Rich Media Technical Support team at [dclk-drmtechnical@google.com](mailto:dclk-drmtechnical@google.com).

About

DoubleClick

For advertisers and publishers who need to reach a target audience, the Google's DoubleClick™ product suite is an advertising platform that maximizes revenue growth and return on advertising spend through a unique and innovative ad targeting process. The experience and innovative spirit at DoubleClick drives a constant evolution of products and solutions, ensuring the best, most effective advertising tools are always at our customers' command.

[www.doubleclick.com](http://www.doubleclick.com)

[www.richmediagallery.com](http://www.richmediagallery.com)

DoubleClick UK : Belgrave House, 76 Buckingham Palace Road, London SW1W 9TQ : Phone: +44 (0)800 912 1344  
[www.doubleclick.co.uk](http://www.doubleclick.co.uk) ©2010 Google Inc. All rights reserved.